UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/808,175	03/23/2004	Cary A. Jardin	XP.002CP4	4827

27189 7590 12/08/2006

PROCOPIO, CORY, HARGREAVES & SAVITCH LLP
530 B STREET
SUITE 2100
SAN DIEGO, CA 92101

EXAMINER
DWIVEDI, MAHESH H

ART UNIT	PAPER NUMBER
2168	

DATE MAILED: 12/08/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/808,175	JARDIN, CARY A.	
	Examiner	Art Unit	
	Mahesh H. Dwivedi	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *3/23/2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. ______.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The disclosure is objected to because of the following informalities: The attorney docket numbers listed on Paragraph 1 should be replaced with application serial numbers and filing dates.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claim 12 recites the limitation "method" in line 1. There is insufficient antecedent basis for this limitation in the claim, as the independent base claim 11 recites a "distributed database system".

Claims 13-20 recite similar language and are rejected as well.

Claim 22 recites the limitation "method" in line 1. There is insufficient antecedent basis for this limitation in the claim, as the independent base claim 21 recites a "system".

Claims 23-24 recite similar language and are rejected as well.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Cruanes et al.** (U.S. Patent 6,954,776) and in view of **Luo et al.** (U.S. Patent 6,804,678).

6. Regarding claim 1, **Cruanes** teaches a method comprising:

A) receiving a database query command at a first node (Column 1, lines 24-29, Column 6, lines 64-67-Column 7, lines 1-14);

B) generating a join table for each of a plurality of processors on said first node in accordance with said database query command (Column 6, lines 64-67-Column 7, lines 1-14);

C) said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node (Column 6, lines 64-67-Column 7, lines 1-14);

The examiner notes that **Cruanes** teaches **"receiving a database query command at a first node"** as "A SQL statement comprises either a query or a combination of a query and data manipulation operations to be performed on a database. The query portion and the data manipulation operations are herein referred to as "operations"" (Column 1, lines 24-28) and "Assume that a join operation is to be performed between source Table S and target Table T" (Column 6, lines 64-65). The

examiner further notes that **Cruanes** teaches **"generating a join table for each of a plurality of processors on said first node in accordance with said database query command"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6). The examiner further notes that **Cruanes** teaches **"said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6).

Cruanes does not explicitly teach:

D) sending a first message having a single copy of said join table from a first shared memory router on said first node to a second shared memory router on a second node;

E) storing said single copy of said join table in a common memory of said second node; and

F) sending a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory.

Luo, however, teaches **"sending a first message having a single copy of said join table from a first shared memory router on said first node to a second shared memory router on a second node"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes…The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), **"storing said single copy of said join table in a common memory of said second node"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), and **"sending a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes…The split vectors perform range portioning, according to one embodiment,

such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 2, **Cruanes** further teaches a method comprising:

A) comparing said single copy of said join table stored in said common memory by each of said plurality of processors on said second node to generate a plurality of intermediate results files (Column 6, lines 64-67-Column 7, lines 1-14).

The examiner notes that **Cruanes** teaches **"comparing said single copy of said join table stored in said common memory by each of said plurality of processors on said second node to generate a plurality of intermediate results files"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6).

Regarding claim 3, **Cruanes** does not explicitly teach a method comprising:

A) sending said plurality of intermediate results files from said second shared memory router to said first shared memory router.

Luo, however, teaches **"sending said plurality of intermediate results files from said second shared memory router to said first shared memory router"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), and "For the non-blocking parallel band join algorithm, in one embodiment, all the join result tuples are computed once, to ensure that a correct join is obtained. Further, the non-blocking parallel band join algorithm is non-blocking, which ensures that intermediate results are available" (Column 12, lines 15-21).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 4, **Cruanes** does not explicitly teach a method comprising:

A) generating a final results file from said plurality of intermediate results files.

Luo, however, teaches **"generating a final results file from said plurality of intermediate results files"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), and "Once the split vectors V_A and V_B are created, a non-blocking parallel band join algorithm simultaneously performs operations on each node 10 using multi-threading" (Column 7, lines 12-16).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 5, **Cruanes** does not explicitly teach a method comprising:

A) executing post-processing operations on said final results file.

Luo, however, teaches **"executing post-processing operations on said final results file"** as "The three steps of the third stage are combined in FIG. 11D" (Column 11, lines 57-58).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 6, **Cruanes** does not explicitly teach a method comprising:

A) wherein said second message comprises a memory pointer.

Luo, however, teaches **"wherein said second message comprises a memory pointer"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 7, **Cruanes** further teaches a method comprising:

A) wherein said portions of said database table are stored by each of said plurality of processors in substantially equal portions (Column 4, lines 61-67-Column 5, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said portions of said database table are stored by each of said plurality of processors in substantially equal portions"** as "Typically, work performance is improved when the slave processes in the shared disk system have equal work to avoid workload skewing. Work skewing occurs when some of the slave processes perform significantly more work than other slave processes. In the present example, since there are a total of 4 partition-pairs, 2 partition-pairs may be assigned to node 304, and 2 partition-pairs to node 310" (Column 4, lines 61-65).

Regarding claim 8, **Cruanes** further teaches a method comprising:

A) wherein said portion of said database table are stored by each of said plurality of processors in substantially equal portions according to a round robin distribution (Column 6, lines 48-60).

The examiner notes that **Cruanes** teaches **"wherein said portion of said database table are stored by each of said plurality of processors in substantially equal portions according to a round robin distribution"** as "During the initial distribution of first-phase partition-pairs, first-phase partition pairs 406, 408 are assigned to node 404 and first-place partition-=pairs 416, 418, are assigned to node 410 for reasons of node affinity. In the next round of distribution, the remaining first=phase partition-pairs 412, 414 are distributed to node 410 because node 410 has available slave processes to operate on the first-phase partition-pairs 412, 414 for the equi-join operation" (Column 6, lines 52-60).

Regarding claim 9, **Cruanes** further teaches a method comprising:

A) wherein said storing of said portions of said database table are stored on a volatile memory of said first and second nodes (Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said storing of said portions of said database table are stored on a volatile memory of said first and second nodes"** as "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Volatile media includes dynamic memory, such as main memory 506" (Column 7, lines 65-67-Column 8, lines 1-3).

Regarding claim 10, **Cruanes** further teaches a method comprising:

A) storing said portions of said database table on a persistent storage device (Column 3, lines 53-55, Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"storing said portions of said database table on a persistent storage device"** as "For the purpose of explanation, it shall be assumed that Tables A and B are stored on persistent storage 302" (Column 4, lines 53-54) and "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Non-volatile media includes, for example, optical or magnetic disks, such as storage device 510" (Column 7, lines 65-67-Column 8, lines 1-2).

Regarding claim 11, **Cruanes** teaches a distributed database system comprising:

A) a first node configured to receive a database query command (Column 1, lines 24-29, Column 6, lines 64-67-Column 7, lines 1-14);

B) a plurality of processors on said first node configured to generate a join table in accordance with said database query command (Column 6, lines 64-67-Column 7, lines 1-14);

C) said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node (Column 6, lines 64-67-Column 7, lines 1-14);

The examiner notes that **Cruanes** teaches **"a first node configured to receive a database query command"** as "A SQL statement comprises either a query or a combination of a query and data manipulation operations to be performed on a database. The query portion and the data manipulation operations are herein referred to as "operations"" (Column 1, lines 24-28) and "Assume that a join operation is to be

performed between source Table S and target Table T" (Column 6, lines 64-65). The examiner further notes that **Cruanes** teaches **"a plurality of processors on said first node configured to generate a join table in accordance with said database query command"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6). The examiner further notes that **Cruanes** teaches **"said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6).

Cruanes does not explicitly teach:

D) a first shared memory router on said first node configured to send a first message having a single copy of said join table;

E) a second shared memory router on a second node configured to receive said first message and store said single copy of said join table in a common memory of said second node; and

F) send a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory.

Luo, however, teaches **"a first shared memory router on said first node configured to send a first message having a single copy of said join table"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes... The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), and "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), **"a second shared memory router on a second node configured to receive said first message and store said single copy of said join table in a common memory of said second node"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), and **"send a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other

of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 12, **Cruanes** further teaches a method comprising:

A) wherein said plurality of processors on said second node are configured to compare said single copy of said join table stored in said common memory and to generate a plurality of intermediate results files (Column 6, lines 64-67-Column 7, lines 1-14).

The examiner notes that **Cruanes** teaches **"wherein said plurality of processors on said second node are configured to compare said single copy of said join table stored in said common memory and to generate a plurality of intermediate results files"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6).

Regarding claim 13, **Cruanes** does not explicitly teach a method comprising:

A) wherein said second shared memory router is further configured to send said plurality of intermediate results files to said first shared memory router.

Luo, however, teaches **"wherein said second shared memory router is further configured to send said plurality of intermediate results files to said first shared memory router"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), and "For the non-blocking parallel band join algorithm, in one embodiment, all the join result tuples are computed once, to ensure that a correct join is obtained. Further, the non-blocking parallel band join algorithm is non-blocking, which ensures that intermediate results are available" (Column 12, lines 15-21)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching

Luo's would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 14, **Cruanes** does not explicitly teach a method comprising:
A) a primary controller on said first node configured to generate a final results file from said plurality of intermediate results files.

Luo, however, teaches **"a primary controller on said first node configured to generate a final results file from said plurality of intermediate results files"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), and "Once the split vectors V_A and V_B are created, a non-blocking parallel band join algorithm simultaneously performs operations on each node 10 using multi-threading" (Column 7, lines 12-16).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of

memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 15, **Cruanes** does not explicitly teach a method comprising:
A) wherein said primary controller is further configured to execute post-processing operations on said final results file.

Luo, however, teaches **"wherein said primary controller is further configured to execute post-processing operations on said final results file"** as "The three steps of the third stage are combined in FIG. 11D" (Column 11, lines 57-58).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 16, **Cruanes** does not explicitly teach a method comprising:
A) wherein said second message comprises a memory pointer.

Luo, however, teaches **"wherein said second message comprises a memory pointer"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm

partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 17, **Cruanes** further teaches a method comprising:

A) wherein said plurality of processors are configured to store portions of said database table in substantially equal portions (Column 4, lines 61-67-Column 5, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said plurality of processors are configured to store portions of said database table in substantially equal portions"** as "Typically, work performance is improved when the slave processes in the shared disk system have equal work to avoid workload skewing. Work skewing occurs when some of the slave processes perform significantly more work than other slave processes. In the present example, since there are a total of 4 partition-pairs, 2 partition-pairs may be assigned to node 304, and 2 partition-pairs to node 310" (Column 4, lines 61-65).

Regarding claim 18, **Cruanes** further teaches a method comprising:

A) wherein said plurality of processors are configured to store portions of said database table in substantially equal portions according to a round robin distribution (Column 6, lines 48-60).

The examiner notes that **Cruanes** teaches **"wherein said plurality of processors are configured to store portions of said database table in substantially equal portions according to a round robin distribution"** as "During the initial distribution of first-phase partition-pairs, first-phase partition pairs 406, 408 are assigned to node 404 and first-place partition-=pairs 416, 418, are assigned to node 410 for reasons of node affinity. In the next round of distribution, the remaining first=phase partition-pairs 412, 414 are distributed to node 410 because node 410 has available slave processes to operate on the first-phase partition-pairs 412, 414 for the equi-join operation" (Column 6, lines 52-60).

Regarding claim 19, **Cruanes** further teaches a method comprising:

A) wherein said plurality of processors are configured to store said portions of said database table on a volatile memory of said first and second nodes (Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said plurality of processors are configured to store said portions of said database table on a volatile memory of said first and second nodes"** as "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Volatile media

includes dynamic memory, such as main memory 506" (Column 7, lines 65-67-Column 8, lines 1-3).

Regarding claim 20, **Cruanes** further teaches a method comprising:

A) wherein said plurality of processors are configured to store said portions of said database table on a persistent storage device (Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said plurality of processors are configured to store said portions of said database table on a persistent storage device"** as "For the purpose of explanation, it shall be assumed that Tables A and B are stored on persistent storage 302" (Column 4, lines 53-54) and "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Non-volatile media includes, for example, optical or magnetic disks, such as storage device 510" (Column 7, lines 65-67-Column 8, lines 1-2).

Regarding claim 21, **Cruanes** teaches a system comprising:

A) a first node having a first shared memory router (Column 1, lines 24-29, Column 6, lines 64-67-Column 7, lines 1-14);

B) a plurality of processors on said first node configured to generate a join table in accordance with said database query command (Column 6, lines 64-67-Column 7, lines 1-14);

C) said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node (Column 6, lines 64-67-Column 7, lines 1-14);

The examiner notes that **Cruanes** teaches **"a first node having a first shared memory router"** as "A SQL statement comprises either a query or a combination of a query and data manipulation operations to be performed on a database. The query portion and the data manipulation operations are herein referred to as "operations"" (Column 1, lines 24-28) and "Assume that a join operation is to be performed between source Table S and target Table T" (Column 6, lines 64-65). The examiner further notes that **Cruanes** teaches **"a plurality of processors on said first node configured to generate a join table in accordance with said database query command"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6). The examiner further notes that **Cruanes** teaches **"said join table being generated from a portion of a database table stored by each of said plurality of processors on said first node"** as "Assume that Target Table T is statically partitioned. A slave process may then be assigned to each partition. In one embodiment of the invention, the join operation is performed by sending each tuple from source Table S only to the group of slave processes that is working on the static partition to which the tuple is mapped" (Column 7, lines 1-6).

Cruanes does not explicitly teach:

D) a first shared memory router on said first node configured to send a first message having a single copy of said join table;

E) a second shared memory router on a second node configured to receive said first message and store said single copy of said join table in a common memory of said second node; and

F) send a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory.

Luo, however, teaches **"a first shared memory router on said first node configured to send a first message having a single copy of said join table"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64), and "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), **"a second shared memory router on a second node configured to receive said first message and store said single copy of said join table in a common memory of said second node"** as "Each node 10

additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7), and **"send a second message to a plurality of processors on said second node indicating the location of said single copy of said join table stored in said common memory"** as "Each node 10 additionally includes a memory 18, to which the tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 22, **Cruanes** does not explicitly teach a method comprising:

A) wherein said first data message comprises a memory pointer.

Luo, however, teaches **"wherein said first data message comprises a memory pointer"** as "Each node 10 additionally includes a memory 18, to which the

tuples 12 may be transferred, such as during a join or other query processing operation" (Column 5, lines 4-7) and "The non-blocking parallel band join algorithm partitions one of the tables such that each tuple of the table ends up on a single node. The algorithm partitions the other of the tables such that some of its tuples end up on two nodes...The split vectors perform range portioning, according to one embodiment, such that tuples 12 with similarly valued attributes, within a range designated by the split vector 15, end up on the same node 10" (Column 5, lines 48-64).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of the cited references because teaching **Luo's** would have allowed **Cruanes's** to provide a method for the dynamic allocation of memory in multi-node join operations to improve efficiency, as noted by **Luo** (Pages Column 4, lines 23-33).

Regarding claim 23, **Cruanes** further teaches a method comprising:

A) wherein said first node is configured to store said first data message on a volatile memory of said first node (Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said first node is configured to store said first data message on a volatile memory of said first node"** as "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Volatile media includes dynamic memory, such as main memory 506" (Column 7, lines 65-67-Column 8, lines 1-3).

Regarding claim 24, **Cruanes** further teaches a method comprising:

A) wherein said first node is configured to store said first data message on a volatile memory of said first node (Column 7, lines 63-67-Column 8, lines 1-8).

The examiner notes that **Cruanes** teaches **"wherein said first node is configured to store said first data message on a volatile memory of said first node"** as "a medium may take many forms, including but not limited to, non-volatile media, volatile media...Non-volatile media includes, for example, optical or magnetic disks, such as storage device 510" (Column 7, lines 65-67-Column 8, lines 1-2).

Conclusion

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Patent 7,092954 issued to **Ramesh** on 15 August 2006. The subject matter disclosed therein is pertinent to that of claims 1-24 (e.g., methods for processing database query commands across multi-processor multi node systems).

U.S. Patent 7,092954 issued to **Zait et al.** on 16 December 2003. The subject matter disclosed therein is pertinent to that of claims 1-24 (e.g., methods for processing database query commands across multi-processor multi node systems).

U.S. Patent 6,564,221issued to **Shatdal** on 13 May 2003. The subject matter disclosed therein is pertinent to that of claims 1-24 (e.g., methods for processing database query commands across multi-processor multi node systems).

Contact Information

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mahesh Dwivedi whose telephone number is (571) 272-2731. The examiner can normally be reached on Monday to Friday 8:20 am – 4:40 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim Vo can be reached (571) 272-3642. The fax number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mahesh Dwivedi

Patent Examiner

Art Unit 2168

Tim Vo

TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

November 30, 2006

Leslie Wong LW

Primary Examiner